

# **PREMIUM AUDIT**

NAME OF PROJECT	Polka Classic
TOKEN BEP20 ADDRESS	0x79Ca48c3F6bfB3eF284b4d4C39cA51286830f9aE
WEBSITE / TELEGRAM	http://polkaclassic.com



DISCLAIMER: PLEASE READ FULL AUDIT



#### **IMPORTANT DISCLAIMER**

NOT RESPONSABLE

Our audit report is not responsible for anything if the project audited turn into a scam.

II. NOT GUARANTEE

Analytix Audit in no way guarantees that a project will not remove liquidity, sell off team tokens, or exit scam.

INFORMATION

Analytix Audit do a research and provides public information about the project in an easy-to-understand format for the common person.

IV.

AUDIT AGREE

Agreeing to an audit in no way guarantees that a team will not remove all liquidity ("Rug Pull"), remove liquidity slowly, sell off tokens, quit the project, or completely exit scam

٧.

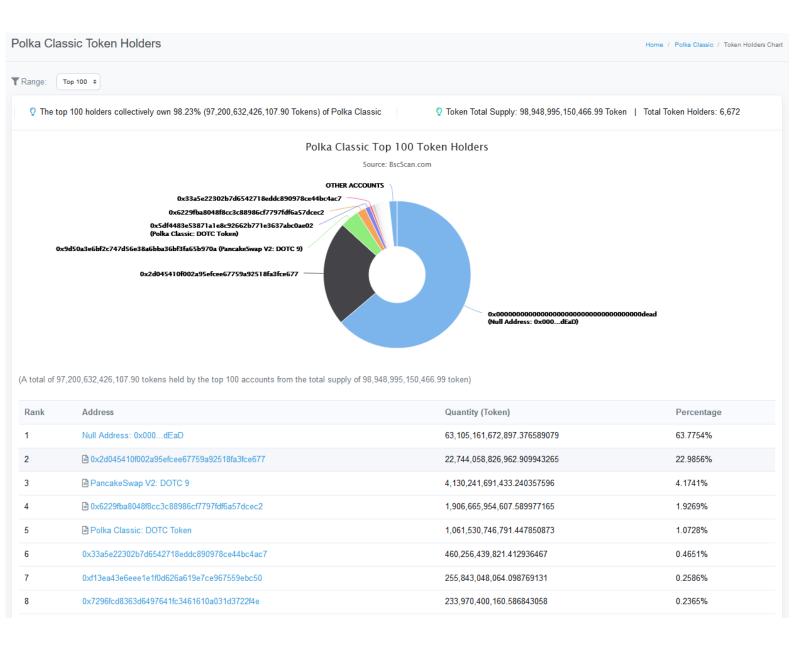
ENCOURAGE INVESTMENT Analytix Audit in no way takes responsibility for any losses, nor does Analytix Audit encourage any speculative investments

۷١.

INFORMATION

The information provided in this audit is for information purposes only and should not be considered investment advice. Analytix Audit does not endorse, recommend, support, or suggest any projects that have been audited.

## **TOKENOMICS - OVERVIEW**



(This comment refers to when the audit was carried out maybe team locked tokens later)

### **CONTRACT CODE - OVERVIEW**

# **QUICK REVIEW**

- Owner can change fees up to 25%
  - Owner can't set Tx Limits

### **Owner Privileges:**

```
* Requirements:

* - the calling contract must have an ETH balance of at least 'value'.

* - the called Solidity function must be 'payable'.

* - the called Solidity function must be 'payable'.

* Available since v3.1._

*/

* function functionCallWithValue(
    address target,
    bytes memory data,
    uint256 value

* jinternal returns (bytes memory) {
    return functionCallWithValue(target, data, value, "Address: low-level call with value failed");

* # @dev Same as {xref-Address-functionCallWithValue-address-bytes-uint256-}['functionCallWithValue'], but

* with 'errorWessage' as a fallback revert reason when 'target' reverts.

* Available since v3.1._

* * _ Available since v3.1._

* function functionCallWithValue(
    address target,
    bytes memory data,
    uint256 value,

* string memory errorMessage

* internal returns (bytes memory) {
    require(address(this).balance >= value, "Address: insufficient balance for call");
    require(isContract(target), "Address: call to non-contract");

* (bool success, bytes memory returndata) = target.call{value: value}(data);
```

# **Issue Checking:**

Ν°	Issue description.	Checking Status.
1	Compiler Errors	Passed
2	Race conditions and Reentrancy. Cross-function race conditions.	Passed
3	Possible delays in data delivery.	Passed
4	Oracle calls.	Passed
5	Front running.	Passed
6	Timestamp dependence.	Passed
7	Integer Overflow and Underflow.	Passed
8	DoS with Revert.	Passed
9	DoS with block gas limit.	Passed
10	Methods execution permissions.	Passed
11	Economy model.	Passed
12	The impact of the exchange rate on the logic.	Passed
13	Private user data leaks.	Passed
14	Malicious Event log	Passed
15	Scoping and Declarations.	Passed
16	Uninitialized storage pointers.	Passed
17	Arithmetic accuracy.	Passed
18	Design Logic.	Passed
19	Cross-function race conditions.	Passed
20	Safe Zeppelin module	Passed
21	Fallback function security.	Passed

#### **Audit Result:**

**PASSED** 

### **WEBSITE - OVERVIEW**



# Domain Registration: 2021-06-04

276 days old

**SSL CERTIFICATE: A** 

(https://www.ssllabs.com)

#### **SOCIAL MEDIA - OVERVIEW**



**POLKACLASSIC** 



CLASSICPOLKA



**POLKACLASSIC** 



**POLKACLASSIC**